Russian Flu Project

Group Members:
Samantha Fisher
Josh Horn
Johanna Pinsirikul
Taylor Sims

Client:
Tom Ewing

CS 4624
Virginia Tech
Blacksburg, VA
Spring 2014

Table of Contents

Executive Summary **2**

User's Manual **3**
- Searching 4
- Links and Other Features 5

Developer's Manual **7**
- Introduction 9
- Getting Started 9
- Apache Solr 10
- Blacklight 13
- Integrating Solr and Blacklight 17
- Site Maintenance 20
- Multimedia 22

Lessons Learned **28**

Acknowledgements **30**

References **31**

Executive Summary

This report includes a detailed description of Dr. Ewing's Russian Flu project. This project integrated two groups: a Translation team and a CS team. The translation team worked to find and translate historical articles documenting the Russian Flu pandemic from 1889 to 1890. The languages consisted of French, German, Spanish, English and Russian. The CS team indexed the article metadata into a searchable website. The desired website allowed a user to search for articles, view a list of corresponding results, and understand the specifics of a given article.

The website was implemented using Solr, an open-source search engine platform and Blacklight, a Ruby on Rails gem. Solr's search features included facets, relevance definitions, spell checking, and synonyms. Blacklight was easily integrated with Solr, in that it displayed the search features and results in a user-friendly format. Blacklight was configured with a faceted search, drop-down menu search, full-text search box, and spell checker. The facets defined for the Russian Flu project included Newspaper Title, Infection Location, Reporting Location, Language, Date, and Keywords. Once a facet is specified, a search is executed, or a combination of the two, a list of corresponding results is displayed. The metadata associated with each article can be viewed as a single document through clicking the Newspaper Title link.

The website also includes multimedia resources tracking the Russian Flu pandemic. An interactive timeline depicts a world map with the flu outbreak and spread at different time intervals. Another multimedia resource is the Google Earth 1889 Russian Flu map overlay. This interactive Google Map allows a user to see all continents and the flu's impact through an 1889 world map.

The Russian Flu website also has a user log in, search history, and article bookmark feature. A user can bookmark an article upon viewing it as a single document. When logged in, a user can view a full list of bookmarked articles. The search history feature saves a list of the user's recent searches, and at the click of a button, the search executes again.

Various problems were encountered throughout the semester, the largest being the lack of a server until April. This kept us from integrating Solr and Blacklight until access to a server was granted. Communication lacked amongst the CS and translation teams, in that the CS team relied on the translation team to format the data correctly in order to index it into Solr.

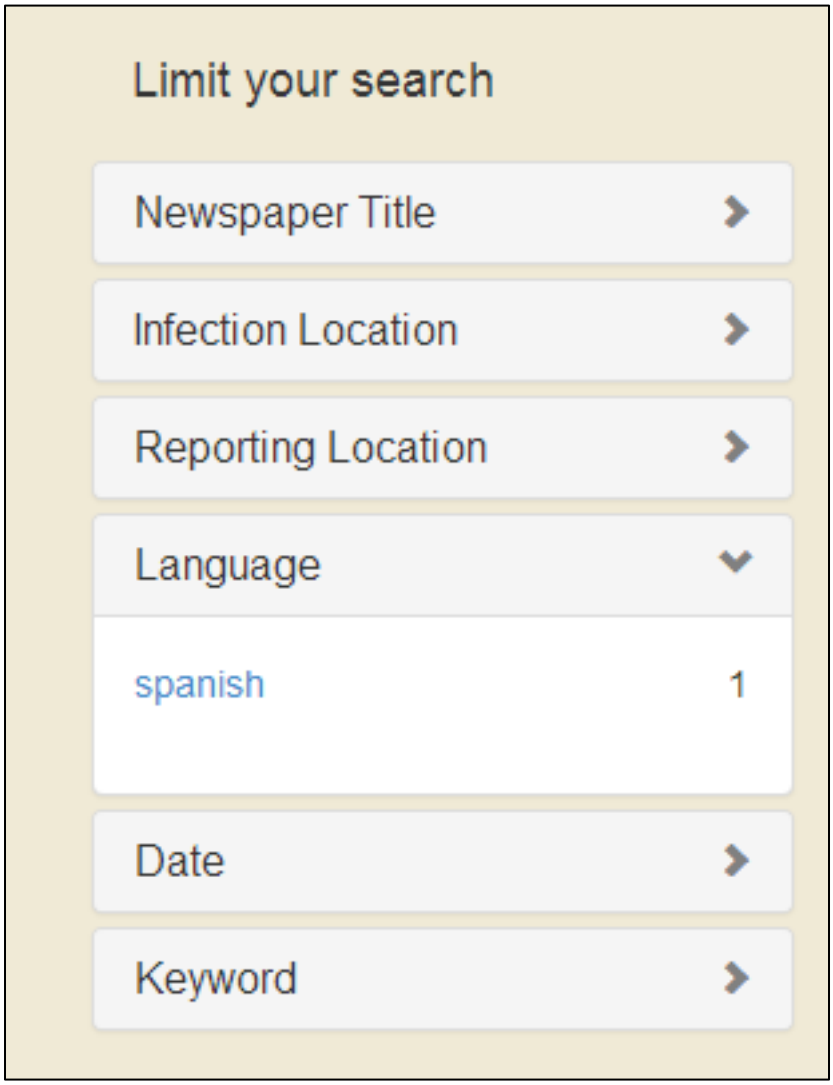
Our client, Dr. Ewing, hopes to continue the Russian Flu project in the future as well as extend the website's functionality. Articles will continue to be translated and article data will continue to be indexed.

Users' Manual

Searching

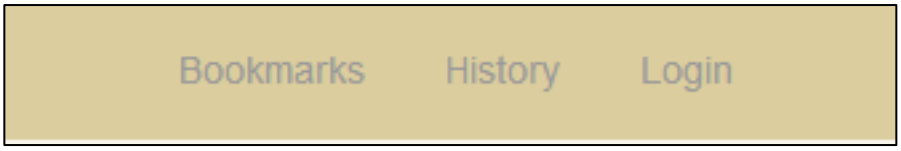
There are two approaches to searching the database for translated articles. The first is to use the search box and drop-down menu. To search within a particular field, use the drop-down menu, and type the corresponding text in the search box.

The second is to use the facet fields. One can also narrow down a search through using a combination of the search box, drop-down menu, and facet fields.
*__Note__: The facets will update to the search parameters.



Links and Other Features

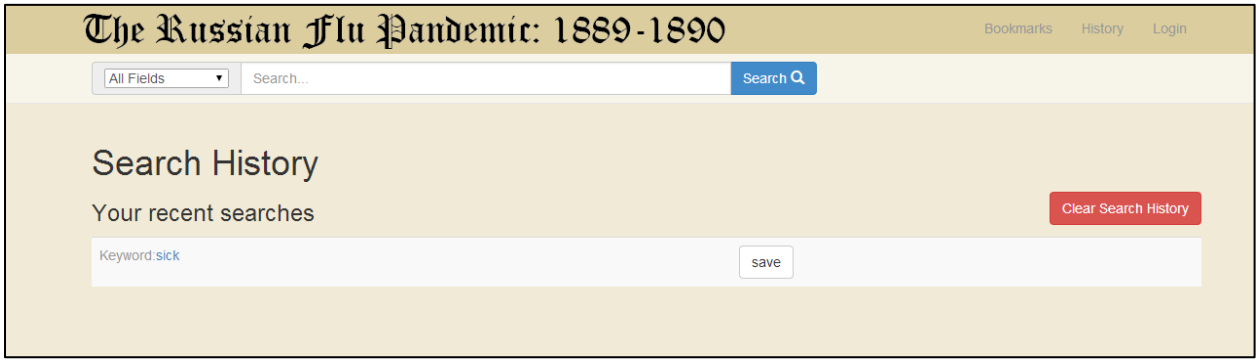
The three links at the top of the webpage are Bookmarks, History, and Login.



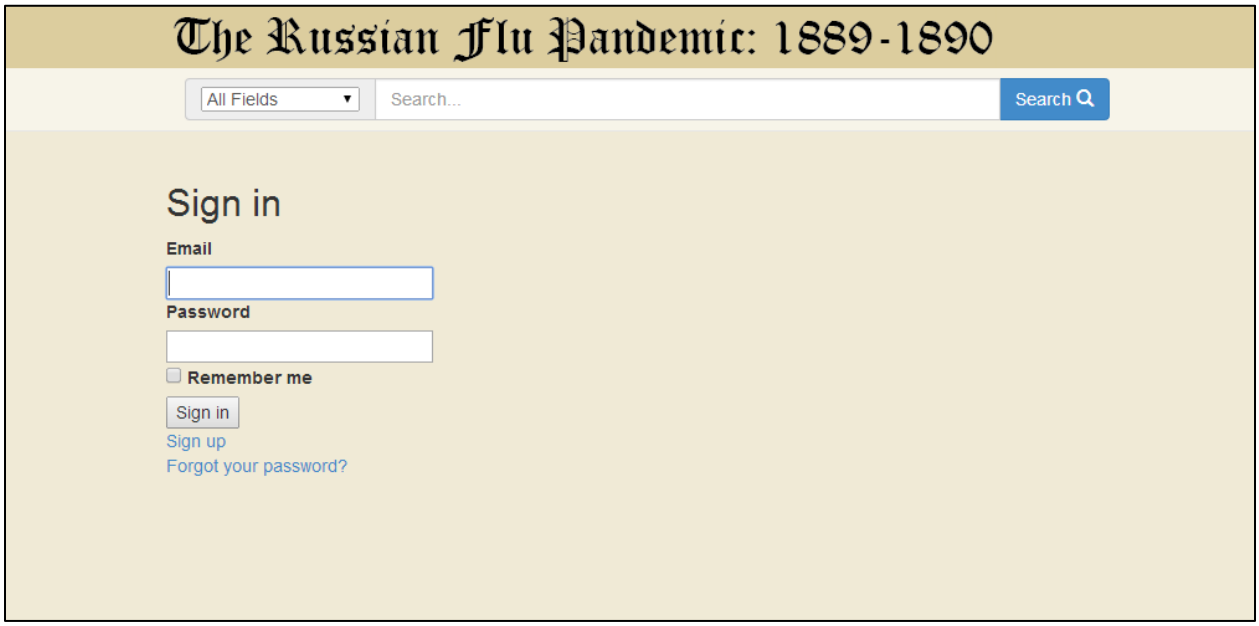
You can bookmark an article by clicking on the bookmark button when you are on its page (which currently you cannot get to).

The Russian Flu Pandemic: 1889-1890
Bookmarks
History
Login
All Fields
Search...
Search
Bookmarks
1 - 3 of 3
10 per page
Clear Bookmarks
Cite
Email
1. Fièvre dengue et grippe, d'après l'épidémie de 1889-1890
In Bookmarks
Newspaper Title: Fièvre dengue et grippe, d'après l'épidémie de 1889-1890
Infection Location: Paris
Original Text: http://gallica.bnf.fr/ark:/12148/bpt6k5442899k/f8.image.r=influenza%20la%20AND%20grippe%20.langEN
2. The Times
In Bookmarks
Newspaper Title: The Times
Infection Location: St. Petersburg
Reporting Location: London
Date: 1889-11-25T00:00:00Z
Translation: Russia. St. Petersburg, Nov. 23. During the week there has been a most unusual case of sickness among all classes in St. Petersburg, which, from the semblance of the symptoms, is supposed to be an epidemic of influenza. It is averred that scarcely a family has escaped, while the numbers of those attacked by it in factories, workshops, and schools are quite phenomenal. In one of the larger mills as many as one hundred workmen at one time have been ill, and parts of the work have had to be suspended. The hospitals are overcrowded, and the doctors complain of too much to do. Most of the journals take it as a text for lamentations over the proverbial insalubrity of the city; others attribute it to the horrible weather of the last few weeks, which has consisted of a series of rapid changes between frost and heat, snow and rain, with heavy clouds and semi-darkness. But if St. Petersburg is the most

You can view your search history by clicking on the History tab. This is what the page looks like:



Clicking the login button allows a user to create an account.



Developer’s Manual

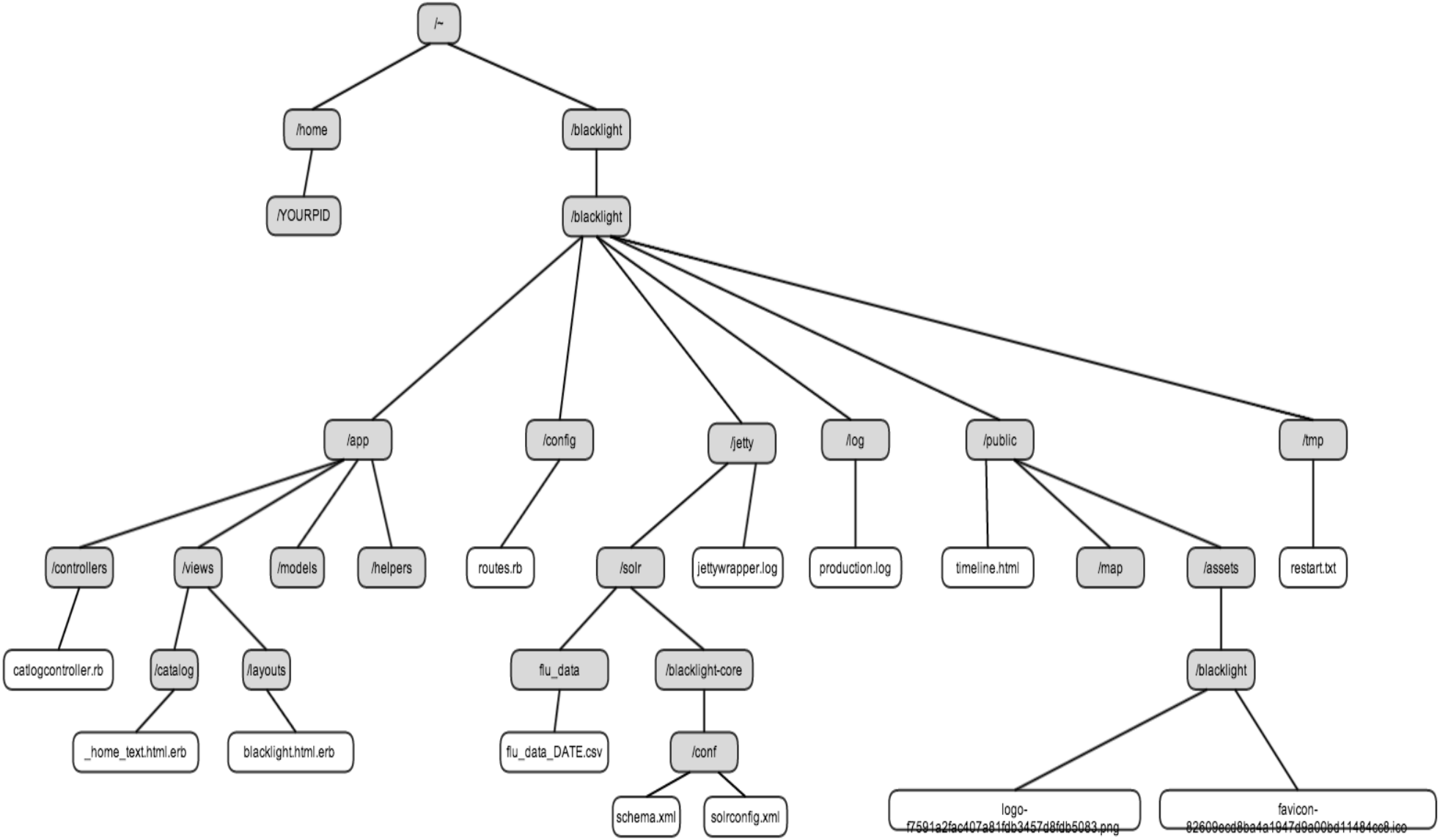


Figure 1. Tree diagram of the file structure on Russian Flu server.

1. Introduction

The Russian Flu website is the product of the integration of Apache Solr and Blacklight, a Ruby on Rails User Interface. This site allows a allow a user to search indexed historical articles documenting the Russian Flu pandemic and view the corresponding search results and multimedia. This manual addresses the set up associated with russianflu.lib.vt.edu as well as the necessary steps to maintain it.

2. Getting Started

2.1 Requirements

In order to use and edit russianflu.lib.vt.edu one must have a stable Internet connection and access to the computer's terminal.
The first thing needed is an installation of Ruby and Rails for Blacklight, and Java for Solr.

The current Solr release requires Java 1.7 or greater, and the current Blacklight release requires Ruby 1.9 or greater, Rails 3.2 and 4.x (although, Rails 4 is recommended), and Java 1.6 or greater.

2.2 Logging In

A user must have an account on the russianflu.lib.vt.edu server in order to access any files associated with the Russian Flu website. Once access is granted, a user can 'ssh' to the Russian flu server by running the command

```
'ssh YOURPID@russianflu.lib.vt.edu'
```

in a terminal. The Russian Flu server can also be accessed through rlogin.
Once on the server, a user will find themselves within their home folder at the full path: /home/YOURPID. The Blacklight and Solr files are located at the full path: /blacklight/blacklight. See Figure 1 for the file configuration within the /blacklight directory.

2.3 Initial Commands

All work done within the Blacklight application should be done as the blacklight user. To become the blacklight user, issue the following command:

```
user:~$  'sudo –u blacklight –s'
```

The Blacklight application is set to **production mode***, so the server is always running. In order to restart the server execute the following commands:

```
user:~$  'sudo –u blacklight service blacklight–jetty stop'
user:~$  'touch tmp/restart.txt'
user:~$  'sudo –u blacklight service blacklight–jetty start'
```

Note: One must be in the /tmp folder in order execute the touch command as-is. Otherwise, redirect touch to the path to the restart.txt file.

* Working in production mode also means that any changes to a file will directly affect the website.

3. Apache Solr

Apache Solr is an open source, full-text search engine platform. Here is a tutorial to set up Solr and configure its capabilities:

http://lucene.apache.org/solr/tutorial.html.

Solr indexes data with a consistent document definition defined in the schema.xml file. The search configuration is an HTTP interface with faceted search, spelling suggestions, and high relevancy results, which is defined in the solrconfig.xml file. Solr resides within the Blacklight application's /jetty directory. See Figure 1 for Solr's file locations.

3.1 Defining Indexed Data

Solr's schema is written in XML and defines the fields and field types for indexed data. See Section 3.3 for the complete schema.xml file.

3.2 Common Tasks

Please see the Section 7, Site Maintenance, for the full steps on the following actions:

3.2.1 Indexing Data

Data is indexed through the computer terminal. Currently, data is uploaded in the format of a tab-separated .csv file through use of the 'curl' command. The process of uploading the translated article data is discussed below in **Section 6.3, Site Maintenance: Indexing Data.**

3.2.2 Clearing the data

Upon altering Solr's schema.xml or solrconfig.xml files, one needs to clear all of the data indexed in Solr. After clearing the data, the data needs to be re-indexed. The changes made to the schema.xml and solrconfig.xml files will then be apparent. See **Section 6.4, Site Maintenance: Clearing/Deleting Data** for the full instructions.

3.3 Schema.xml

Figure 2 depicts the full schema.xml used to index the translated article data. The first XML tag, fields, describes the list of fields to be stored in Solr. The final XML tag, types, described the type of each indexed field, i.e. The Newspaper_Title field is a text_general, while the Date field is a date.

```
<schema name="example" version="1.5">
    <fields>
        <field name="_version_" type="long" indexed="true" stored="true" required="false"/>
        <field name="id" type="text_general" indexed="true" stored="true" required="true" multiValued="false"/>
        <field name="Newspaper_Title" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Infection_Location" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Infect_Lat_Long" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Reporting_Location" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Report_Lat_Long" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Date" type="date" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Link" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Translation" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Language" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Tags_Keywords" type="text_general" indexed="true" stored="true" required="false" multiValued="true" />
        <field name="Image_Link" type="text_general" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="Translator" type="string" indexed="true" stored="true" required="false" multiValued="false" />
        <field name="collector" type="string" indexed="true" stored="false" multiValued="true"/>

        <dynamicField name="*_facet" type="string" indexed="true" stored="false" multiValued="false"/>

        <copyField source="Newspaper_Title" dest="collector"/>
        <copyField source="Infection_Location" dest="collector"/>
        <copyField source="Infect_Lat_Long" dest="collector"/>
        <copyField source="Reporting_Location" dest="collector"/>
        <copyField source="Report_Lat_Long" dest="collector"/>
        <copyField source="Date" dest="collector"/>
        <copyField source="Link" dest="collector"/>
        <copyField source="Translation" dest="collector"/>
        <copyField source="Language" dest="collector"/>
        <copyField source="Tags_Keywords" dest="collector"/>
        <copyField source="Image_Link" dest="collector"/>
    </fields>

    <copyField source="*_facet" dest="collector"/>
    <uniqueKey>id</uniqueKey>
    <defaultSearchField>collector</defaultSearchField>
    <solrQueryParser defaultOperator="OR"/>

    <types>
        <fieldType name="string" class="solr.StrField" sortMissingLast="true" />
<!--    <fieldType name="text_general" class="solr.TextField" sortMissingLast="true" />-->
        <fieldType name="text_general" class="solr.TextField" positionIncrementGap="100">
                 <analyzer type="query">
                        <tokenizer class="solr.StandardTokenizerFactory"/>
                        <filter class ="solr.LowerCaseFilterFactory"/>
                        <filter class ="solr.WordDelimiterFilterFactory" preserveOriginal="1"/>
                </analyzer>
        </fieldType>
        <fieldType name="date" class="solr.TrieDateField" precisionStep="0" positionIncrementGap="0"/>
        <fieldType name="long" class="solr.TrieLongField" precisionStep="0" positionIncrementGap="0"/>
        <fieldType name="text_en" class="solr.TextField" positionIncrementGap="100">
            <!-- lots of details -->
        </fieldType>
        <fieldType name="currency" class="solr.CurrencyField" precisionStep="8" defaultCurrency="USD" currencyConfig="currency.xml" />
    </types>
</schema>
```

Figure 2. Schema.xml file for Russian Flu website.

3.4 Solrconfig.xml

Solr's search capabilities are defined within the **solrconfig.xml** file (Figures 3 - 6). Currently, there are two types of searches defined for the Russian Flu website: Default search and Document search. The default search, defined as "search" in solrconfig.xml, represents all capabilities on the homepage including default search result fields (Figure 3) and facet fields (Figure 4). Document search is use to display a single record (Figure 5).

```
<str name="fl">
  Newspaper_Title,
  Infection_Location,
  Reporting_Location,
  Date,
  Link,
  Language,
  Tags_Keywords
</str>
```

Figure 3. Default search result fields

```
<str name="facet">true</str>
<str name="facet.mincount">1</str>
<!-- str name="facet.limit">10</str -->
<str name="facet.field">Newspaper_Title</str>
<str name="facet.field">Infection_Locationt</str>
<str name="facet.field">Reporting_Location</str>
<str name="facet.field">Language</str>
<str name="facet.field">Date</str>
<str name="facet.field">Tags_Keywords</str>
```

Figure 4. Facet fields in the solrconfig.xml.

```
</requestHandler>

<!-- for requests to get a single document; use id=666 instead of q=id:666 -->
<requestHandler name="document" class="solr.SearchHandler" >
  <lst name="defaults">
    <str name="echoParams">all</str>
    <str name="fl">*</str>
    <str name="rows">1</str>
    <str name="q">{!raw f=id v=$id}</str> <!-- use id=666 instead of q=id:666 -->
  </lst>
</requestHandler>
```

Figure 5. Document view search handler in solrconfig.xml.

4. Blacklight

Blacklight is an open source, Ruby on Rails user interface for Apache Solr. Backlight uses Solr's search engine to query and display indexed data.
Blacklight includes the following features:

- Faceted browsing,
- Relevance based searching,
- Bookmarkable items,
- Permanent URLs for every item,
- User tagging of items

4.1 Setting Up Blacklight

To generate a Blacklight application, please refer to Blacklight's Quickstart Guide:

https://github.com/projectblacklight/blacklight/wiki/Quickstart.

This tutorial will provide and index sample data. See Figure 6 for the resulting (default) Blacklight application.

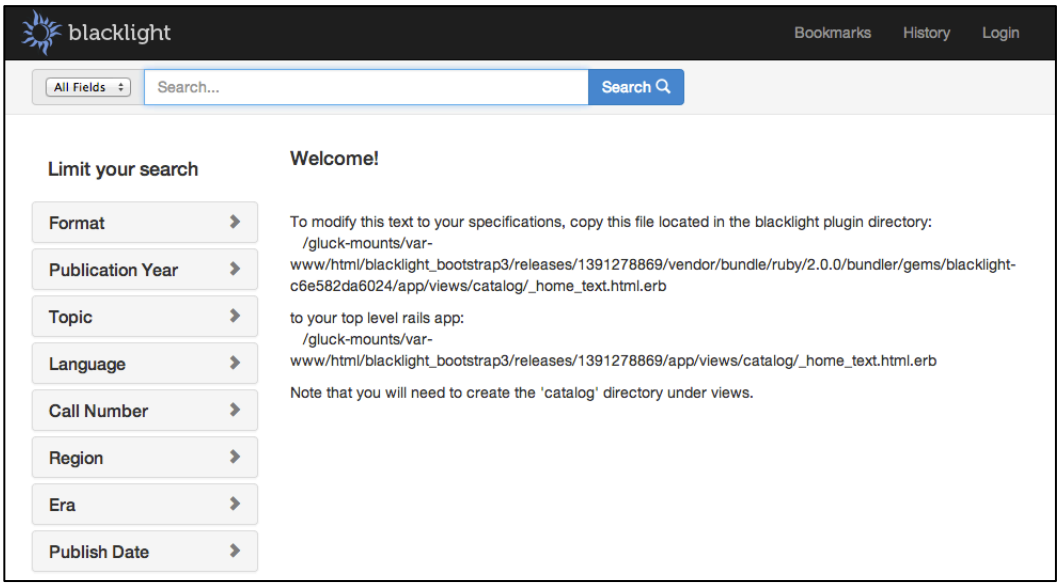


Figure 6. Blacklight's demo site.

4.2 Customizing Blacklight

All customization of the Blacklight instance should be done within the application, i.e. within the /blacklight/blacklight directory. Upon generating the Blacklight application, an instance of Solr including the jetty server is included.

Configuration of the Solr fields and Blacklight visualization including facets, index fields, search parameters, and show fields will go in the **catalog_controller.rb** file. See Figure 1 for the location of this file. For more information on how to tailor the catalog_controller, please refer to the Configuring and Customizing Blacklight section in the Quickstart Guide:

https://github.com/projectblacklight/blacklight/wiki/Configuring-and-Customizing-Blacklight.

For the purpose of the Russian Flu project, our client approved the default layout (Figure 7). Apart from the changes made to the search and facets, we only needed to change the color scheme. This can be done in the .css files provided within the /public/assets directory (Figure 1). The logo image and favicon (an icon associated with a URL that is variously displayed, as in a browser's address bar or next to the site name in a bookmark list) were changed for the website.

CSS is a style guide that formats the different html object ids given in the site upon creation. Please refer to the CSS tutorial:

http://www.w3schools.com/css/default.asp

4.2.1 Changing the favicon and title of the webpage

Figure 7 depicts the current favicon and title.

Figure 7. Current favicon and title.

To change the favicon* replace favicon-82609ecd8ba4a1947d9a00bd11484cc8.ico in the blacklight/blacklight/public/assets directory.

* **Note**: The new icon must be named the same and the size must be 16x16, 32x32, 48x48, or 64x64 pixels.

To change the title, change "Russian Flu" in the line:

<title><%= "Russian Flu" %></title>

in blacklight.html.erb in the blacklight/blacklight/app/views/layouts directory (Figure 1).

4.2.2 Changing the header image

Figure 8 displays the current header image.

Figure 8. Russian Flu site logo.

To change the current header*, replace

logo-f7591a2fac407a81fdb3457d8fdb5083.png

in the /blacklight/blacklight/public/assets/blacklight directory.
*__Note__: the new header image must be named the same

4.2.3 Adding images to the site

Place the desired image in the blacklight/blacklight/app/assets/images directory (Figure 1). To reference the image in one of the .html.erb files use it like:

<%=image_tag("LePetit2.jpg", :width => 650, :height => 310)%>

Blacklight takes care of the image's full path.

4.2.4 Changing the content of the homepage

Figure 9 depicts the current content on the Russian Flu homepage.

Welcome!
Welcome to the Russian Flu
This project explores the relationship between the spread of disease and the spread of information during the "Russian Flu" epidemic, 1889-1890. The research will be done by a team of Virginia Tech undergraduates working in collaboration with a historian. This site will provide updates on the project, resources, and findings.
Check out our Blog!
Russian Flu Blog
Multimedia
Timeline
Overall Flu Spread MAP

Figure 9. Current Russian Flu homepage content.

To change the homepage content, edit the **_home_text.html.erb** in the blacklight/blacklight/app/views/catalog directory (Figure 1).

Figures 10 and 11 display the resulting Russian Flu website.

The Russian Flu Pandemic: 1889-1890
Bookmarks
History
Login
All Fields
Search...
Search
Limit your search
Newspaper Title
Infection Location
Reporting Location
Language
Date
Keyword
Welcome!
Welcome to the Russian Flu
This project explores the relationship between the spread of disease and the spread of information during the "Russian Flu" epidemic, 1889-1890. The research will be done by a team of Virginia Tech undergraduates working in collaboration with a historian. This site will provide updates on the project, resources, and findings.
Check out our Blog!
Russian Flu Blog
Multimedia

Figure 10. Final Russian Flu homepage.

Figure 11. Final Russian Flu search results page.

A few other examples of websites using Blacklight like AGNIC, ALICE, Columbia University, etc. can be found here:

https://github.com/projectblacklight/blacklight/wiki/Examples

5. Integrating Solr and Blacklight

5.1 Facet Search

Using the field names defined in **schema.xml**, the following images depict the implementation of faceted search.

```
config.add_facet_field 'Newspaper_Title', :label => 'Newspaper Title'
config.add_facet_field 'Infection_Location', :label => 'Infection Location'
config.add_facet_field 'Reporting_Location', :label => 'Reporting Location'
config.add_facet_field 'Language', :label => 'Language', :limit => true
config.add_facet_field 'Date', :label => 'Date', :date => { :format => :short }
config.add_facet_field 'Tags_Keywords', :label => 'Keyword'
```

Figure 12. Code in the catalog_controller.rb for facet display on website.

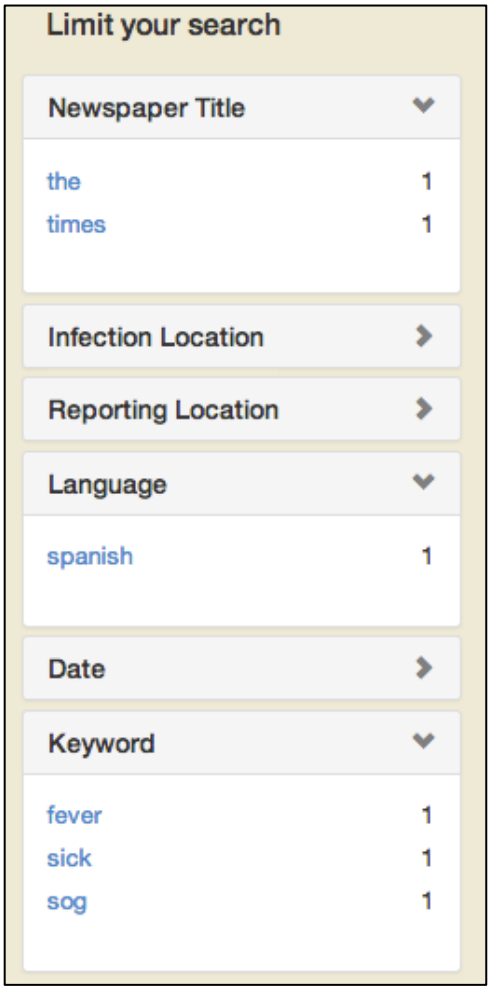


Figure 13. Facet display on site.

```
<str name="facet">true</str>
<str name="facet.mincount">1</str>
<!-- str name="facet.limit">10</str -->
<str name="facet.field">Newspaper_Title</str>
<str name="facet.field">Infection_Locationt</str>
<str name="facet.field">Reporting_Location</str>
<str name="facet.field">Language</str>
<str name="facet.field">Date</str>
<str name="facet.field">Tags_Keywords</str>
```

Figure 14. Code in solrconfig.xml defining facet fields.

5.2 Drop-down Menu

Using the field names defined in **schema.xml**, the following images depict the implementation of the drop-down menu. Figures 16 and 17 must be included in the solrconfig.xml file in order for the drop-down menu fields to appear.

Figure 15. List of defined drop-down search fields.

```
<str name="qf">
  Newspaper_Title^100000
  Tags_Keywords^50000
  Translation^25000
  Infection_Location^10000
  Reporting_Location^5000
  Language^2500
  Date
  collector
</str>
<str name="pf">
  Newspaper_Title^1000000
  Tags_Keywords^500000
  Translation^250000
  Infection_Location^100000
  Reporting_Location^50000
  Language^25000
  Date^100
  collector^10
</str>
```

Figure 16. In solrconfig.xml: Specifies fields to include in default search fields.

```
<str name="newspaper_qf">
  Newspaper_Title^200
</str>
<str name="newspaper_pf">
  Newspaper_Title^200
</str>
<str name="location_qf">
  Infection_Location^50000
  Reporting_Location^25000
</str>
<str name="location_pf">
  Infection_Location^500000
  Reporting_Location^250000
</str>
<str name="language_qf">
  Language
</str>
<str name="language_pf">
  Language
</str>
```

Figure 17. In solrconfig.xml: Specifies fields to include in drop-down menu.

5.3 Results List

The fields to display in the results view can be configured within the catalog_controller.rb file. Figure 18 depicts the commands necessary for the fields in the results view in Figure 19.

```
# solr fields to be displayed in the index (search results) view
#   The ordering of the field names is the order of the display
config.add_index_field 'Newspaper_Title', :label => 'Newspaper Title'
config.add_index_field 'Infection_Location', :label => 'Infection Location'
config.add_index_field 'Reporting_Location', :label => 'Reporting Location'
config.add_index_field 'Date', :label => 'Date', :date => { :format => :short }
config.add_index_field 'Language', :label => 'Language'
config.add_index_field 'Tags_Keywords', :label => 'Keywords'
config.add_index_field 'Translation', :label => 'Translation'
```

Figure 18. Defines fields to show in results view in catalog_controller.rb.

Figure 19. Results view of the Russian Flu website.

5.4 Document View

A single document can be viewed by clicking on the Newspaper Title link (Figure 19). The document view can be configured in the solr.xml file (Figure 20) and the catalog_controller.rb file (Figure 21).

```
</requestHandler>

<!-- for requests to get a single document; use id=666 instead of q=id:666 -->
<requestHandler name="document" class="solr.SearchHandler" >
  <lst name="defaults">
    <str name="echoParams">all</str>
    <str name="fl">*</str>
    <str name="rows">1</str>
    <str name="q">{!raw f=id v=$id}</str> <!-- use id=666 instead of q=id:666 -->
  </lst>
</requestHandler>
```

Figure 20. Document view search handler in solrconfig.xml.

```
# solr fields to be displayed in the show (single result) view
#   The ordering of the field names is the order of the display
config.add_show_field 'Newspaper_Title', :label => 'Newspaper Title'
config.add_show_field 'Infection_Location', :label => 'Infection Location'
config.add_show_field 'Reporting_Location', :label => 'Reporting Location'
config.add_show_field 'Date', :label => 'Date', :date => true
config.add_show_field 'Language', :label => 'Language'
config.add_show_field 'Tags_Keywords', :label => 'Keywords'
config.add_show_field 'Translation', :label => 'Translation'
config.add_show_field 'Image_Link', :label => 'Image Link'
```

Figure 21. Document fields defined in catalog_controller.rb.

6. Site Maintenance

The sections above discussed the initial creation and implementation of the current Russian Flu website. Currently, the Russian Flu website needs to be maintained and expanded per the client's request. The following sections describe common scenarios and commands needed to maintain the website:

6.1 Edit ONLY as the Blacklight User

Developers should only make changes to the Blacklight application (any file in the /blacklight/blacklight directory) as the Blacklight User. Upon accessing the Russian Flu server, a developer's username is their PID. To become the Blacklight user execute:

```
user:~$ 'sudo –u blacklight –s'
```

6.2 Stop/Start Server

Upon changing files in the Blacklight application, a developer must restart the server for the updates to appear.

To stop the server:

```
blacklight:~$ 'sudo –u blacklight service blacklight–jetty
                                    stop'
```

To restart Phusion Passenger:

```
blacklight:~$ 'touch tmp/restart.txt'
```

To start the server:

```
blacklight:~$ 'sudo –u blacklight service blacklight–jetty
start'
```

6.3 Indexing Data

As discussed in the **Section 3.2, Apache Solr: Common Tasks**, the data to be searched and viewed on the website must be indexed in Solr. Below are the steps to upload data:

1. Download the translation excel spreadsheet on Google Drive as a tab-separated values (.tsv) file.
2. Secure-copy the .tsv file from your computer to the server.

```
user:~$ 'scp FILENAME YOURPID@russianflu.lib.vt.edu:'
```

3. SSH into your Russian Flu directory.

```
user:~$ 'ssh YOURPID@russianflu.lib.vt.edu'
```

4. Copy the .tsv file to the /solr directory.

```
blacklight:~/home/user$ 'cp FILENAME
../../blacklight/blacklight/jetty/solr'
```

5. Go to the /solr directory.

```
blacklight:~/home/user$ 'cd
../../blacklight/blacklight/jetty/solr'
```

6. Execute the following command to upload the data:

```
blacklight:~/blacklight/blacklight/jetty/solr$ curl
'http://localhost:8983/solr/update/csv?commit=true&s
eparator=%09&stream.file=solr/FILENAME.csv&stream.co
ntentType=application/csv;charset=utf-8'
```

6.4 Clearing/Deleting Data

As discussed in **Section 3.2, Apache Solr: Common Tasks**, the data to be searched and viewed on the website must be altered in Solr. If the schema.xml or solrconfig.xml file is changed, the indexed data must be dropped and re-indexed for changes to appear. Below are the steps to drop data:

1. Execute the following commands:

```
blacklight:~$ curl http://localhost:8983/solr/update
-H "Content-type: text/xml" --data-binary
'<delete><query>*:*</query></delete>'
```

```
blacklight:~$  curl http://localhost:8983/solr/update
-H "Content-type: text/xml" --data-binary
'<commit/>'
```

2. Now follow the instructions in **Section 6.3** above to re-index the data.

7. Multimedia

7.1 Map

The map that shows the overall Russian Flu spread was made using the MapTiler software (Figure 22). This software allows the use of an image to be overlaid on top of a Google Earth globe. MapTiler must be downloaded onto a local machine.

We needed to use this software to allow us to correctly represent country borders, since we were working with data and images from a different decade.

We were provided with an 1889 map to overlay on top of the globe. Using Photoshop, this map was altered to only contain the important pieces of the image.

The software was then used to upload the image and recognize certain parts on the map to be able to overlay the image correctly (Figure 24).

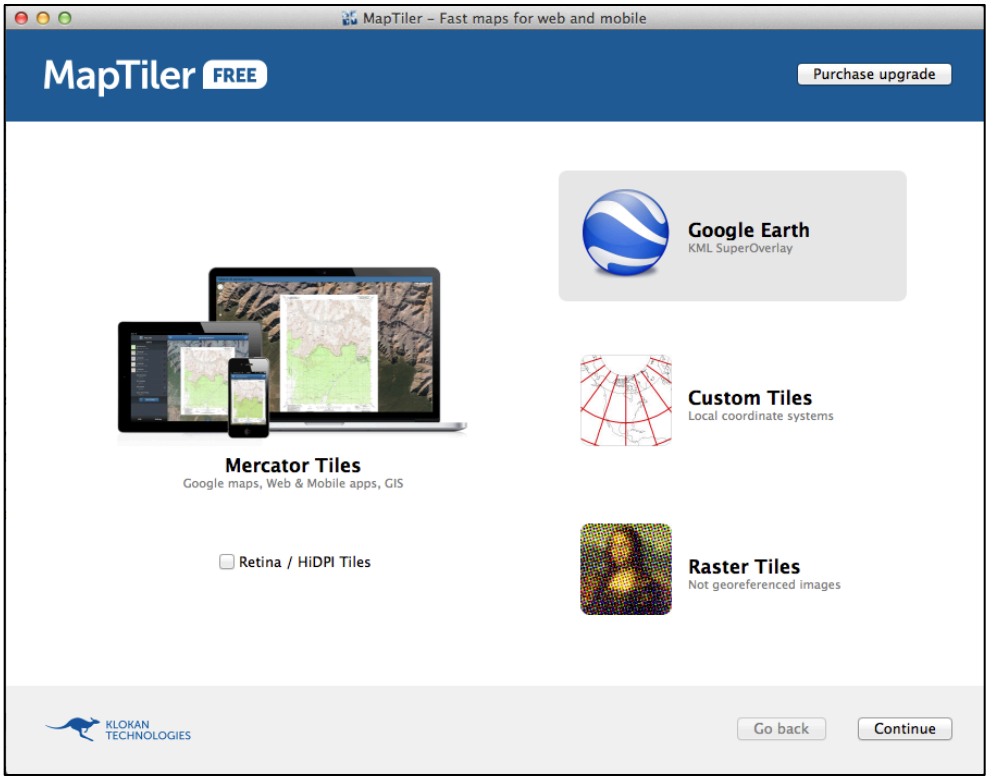


Figure 22. MapTiler Application used to create 1889 Russian Flu Overlay.

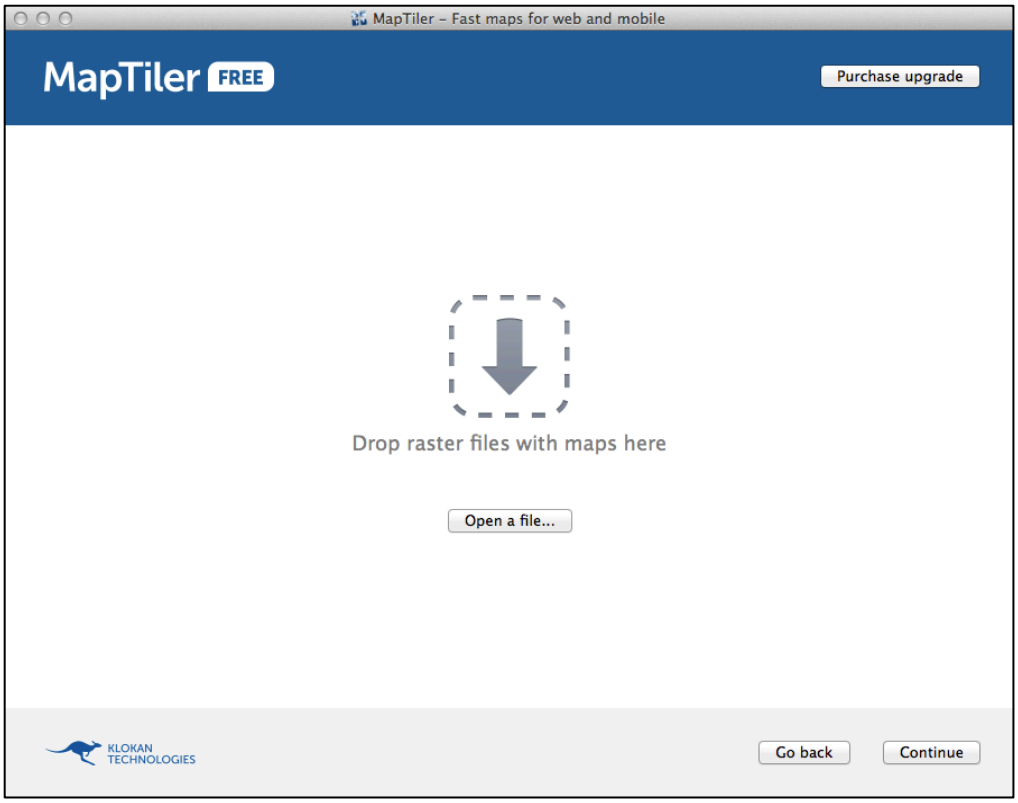


Figure 23. Select the 1889 Russian Flu map image to overlay.

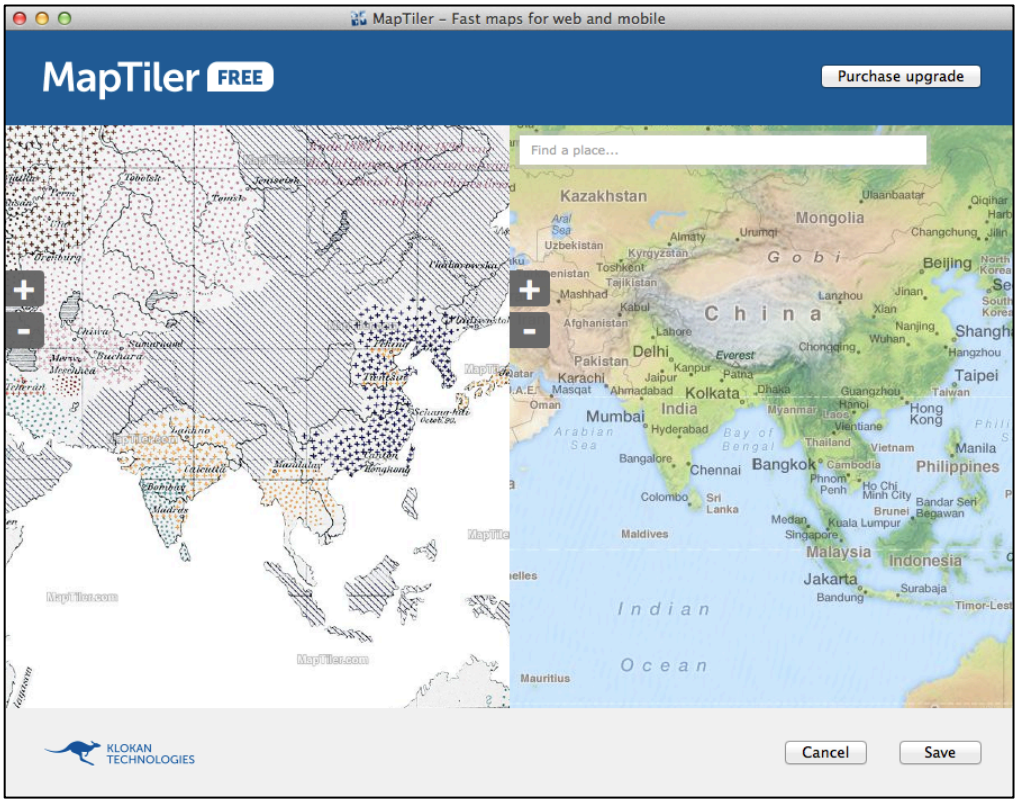


Figure 24. 1889 Russian Flu Map image and current Google Maps compared.

This YouTube video shows how to overlay and image over a map:

https://www.youtube.com/watch?v=eJxdCe9CNYg

To download MapTiler, see some of the features and tutorials:

http://www.maptiler.com

The final result is displayed in Figure 25. See Figure 1 for the file location of the map.

Figure 25. Final Result from MapTiler Application with 1889 Map overlayed.

7.2 Timeline

For the creation of the timeline, we used the open source tool TimelineJS created by Knight Lab.

TimelineJS allows you to easily create any timeline using a Google Doc spreadsheet containing the desired information. There are many different ways you can represent your timeline, and many different types of information that can be included.

Steps to creating a timeline (more detail in TimelineJS page):

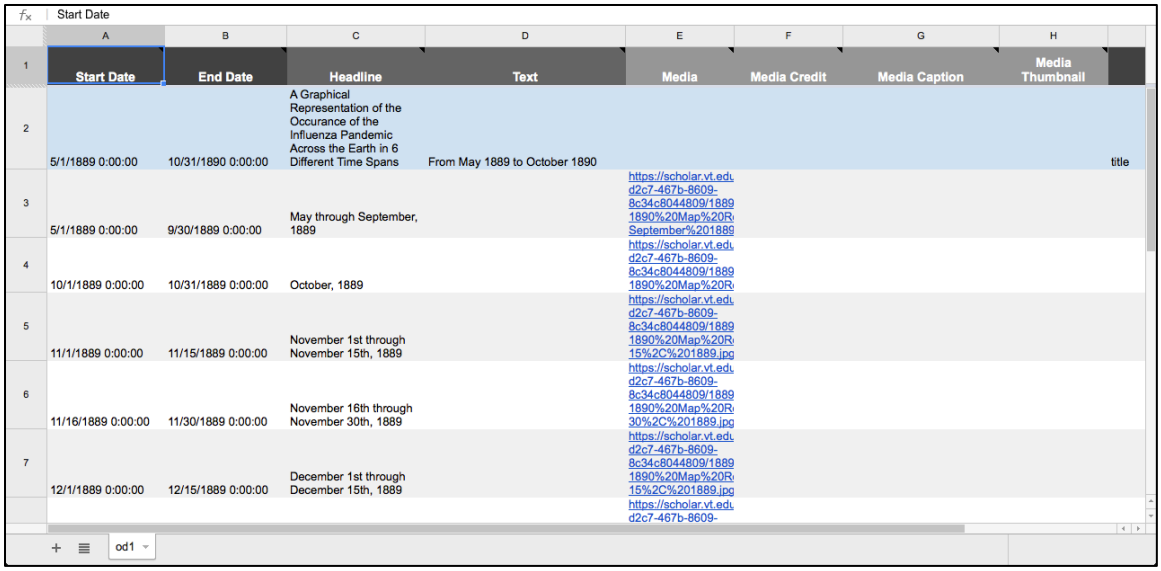
1. Create your spreadsheet from a template provided
2. Publish spreadsheet to the web
3. Copy spreadsheet URL into the box provided in the page
4. Embed the code provided into the website

This YouTube video tutorial shows you how to create your timeline:

https://www.youtube.com/watch?v=vAWbm4gF9lU

For the Russian Flu project, we wanted a time-lapse of how the flu had spread, so we only decided to include images. But YouTube videos can also be used, and text included accompanying the images.

The Russian Flu spreadsheet for TimelineJS is displayed in Figure 26.



	A	B	C	D	E	F	G	H
1	Start Date	End Date	Headline	Text	Media	Media Credit	Media Caption	Media Thumbnail
2	5/1/1889 0:00:00	10/31/1890 0:00:00	A Graphical Representation of the Occurance of the Influenza Pandemic Across the Earth in 6 Different Time Spans	From May 1889 to October 1890				title
3	5/1/1889 0:00:00	9/30/1889 0:00:00	May through September, 1889		https://scholar.vt.edu d2c7-467b-8609- 8c34c8044809/1889 1890%20Map%20R September%201889			
4	10/1/1889 0:00:00	10/31/1889 0:00:00	October, 1889		https://scholar.vt.edu d2c7-467b-8609- 8c34c8044809/1889 1890%20Map%20R			
5	11/1/1889 0:00:00	11/15/1889 0:00:00	November 1st through November 15th, 1889		https://scholar.vt.edu d2c7-467b-8609- 8c34c8044809/1889 1890%20Map%20R 15%2C%201889.jpg			
6	11/16/1889 0:00:00	11/30/1889 0:00:00	November 16th through November 30th, 1889		https://scholar.vt.edu d2c7-467b-8609- 8c34c8044809/1889 1890%20Map%20R 30%2C%201889.jpg			
7	12/1/1889 0:00:00	12/15/1889 0:00:00	December 1st through December 15th, 1889		https://scholar.vt.edu d2c7-467b-8609- 8c34c8044809/1889 1890%20Map%20R 15%2C%201889.jpg			
					https://scholar.vt.edu d2c7-467b-8609-			

Figure 26. Google Doc spreadsheet for TimelineJS.

Figures 27 and 28 depict the final Timeline. See Figure 1 for the file location of the timeline.

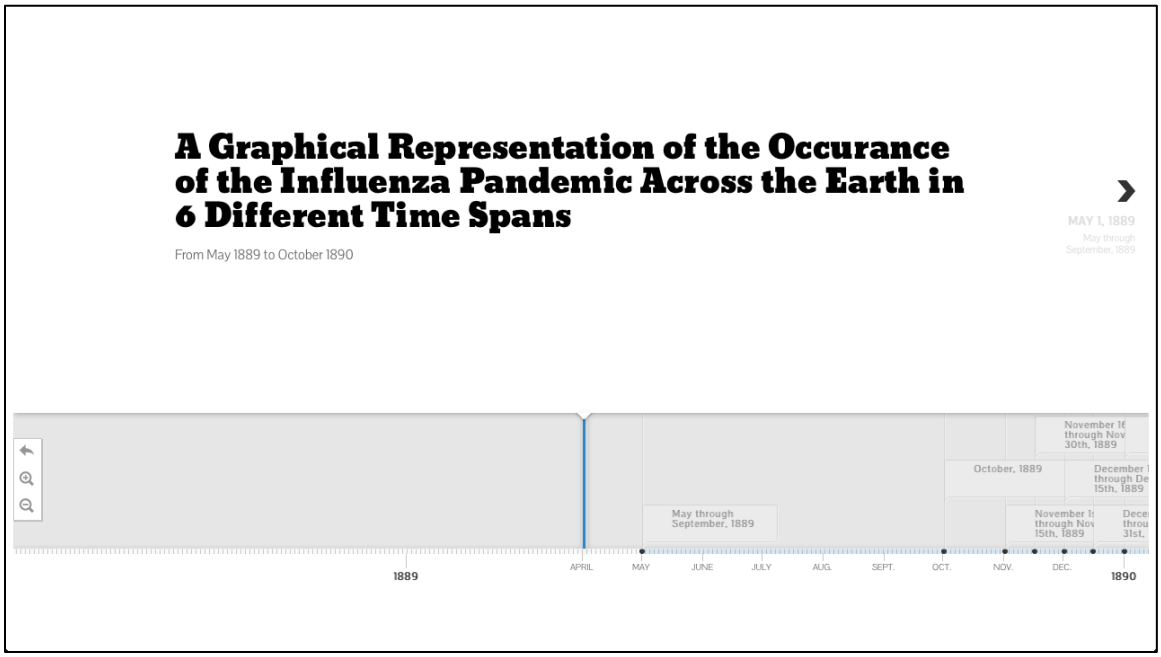


Figure 27. Final Timeline front page.

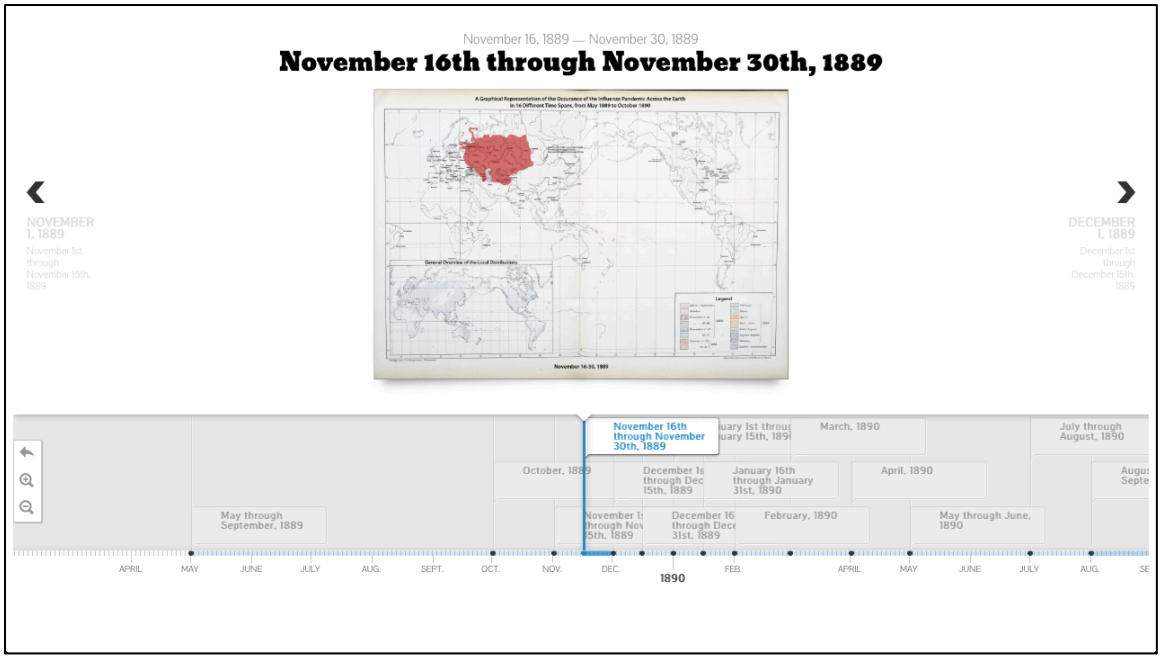


Figure 28. Final Timeline structure.

For more information about TimelineJS and other views:

http://timeline.knightlab.com

7.3 Other Multimedia Resources

StoryMapJS

Knight Lab provides another tool, StoryMapJS, which allows one to add information to a map and go forward chronologically/geographically. Figures 29 and 30 depict a working version of StoryMapJS. This will be a resource for expansion in the future.

Figure 29. StoryMapJS displays chronological information.

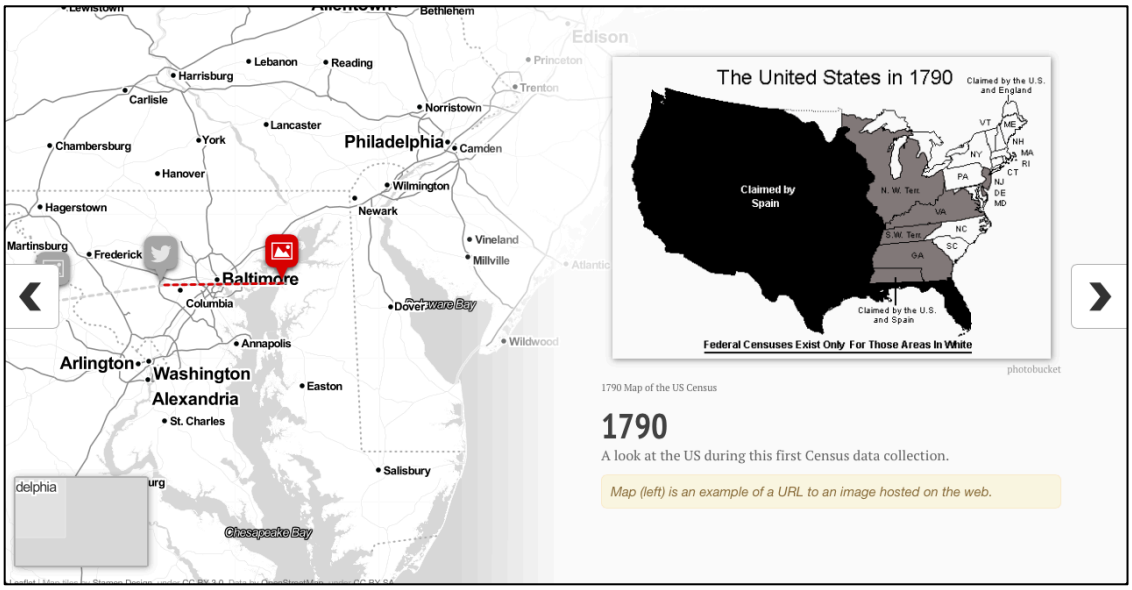


Figure 30. StoryMapJS at specific data point.

More information on this resource:

http://storymap.knightlab.com

Lessons Learned

Timeline

Our client signed off on our project on February 3rd, and we spent the following two weeks researching tools to best suit our client's requests. We researched various ways to store and search data and decided Solr would be the most effective search mechanism. We then had to decide on a front-end to use that integrated with Solr. We settled on using Blacklight, an open source Ruby on Rails gem that provides a discovery for any Solr index.

In March, we created a prototype of the user interface's design and function. Another major decision regarded the configuration of the searching. We decided to implement a faceted search option and began organizing the fields. The final list of facet fields was: Newspaper Title, Infection Location, Reporting Location, Language, Date, and Keywords. The majority of March consisted of working with the translation team associated with the project. We had to explain the formats for each data field in order to index the articles into Solr and ensure that every required field had a value and the format of each category was valid and consistent. A major alteration in the formats pertained to the storage of the date field. Solr expected the dates in the format, YYYY-MM-DDTHH:MM:SSZ (YYYY-MM-DDT00:00:00Z); therefore, the translation team had to update their articles.

April 1st marked the setup completion of the Russian Flu server. This allowed us to stop working on our local machines and move files onto the library server. On April 28th, the Russian Flu project team gave a presentation showcasing accomplishments thus far and future plans. Our team spent the latter part of the semester troubleshooting various problems and fixing bugs, some of which were not solved.

Problems and Solutions

Throughout the course of our project, we encountered various problems, most of which we found solutions. The first problem encountered pertained to the searching and storage of article metadata. We researched many databases including SQL and mongoDB. After meeting with Dr. Fox, we decided the best implementation included using Apache Solr. Upon selecting Solr as a back-end, we needed to find a front-end user interface in which to integrate Solr. From our research we concluded that Blacklight would suffice, in that there would not be a steep learning curve. We also chose Blacklight, because some of group members were familiar with Ruby on Rails.

Our next problem was getting access to server in which to host our website. On April 1st, we were granted access to a library server, which allowed us to begin integrating Solr and Blacklight in a remote location. Unfortunately, gaining access to a server

this late in the semester hindered the amount of work we could accomplish and problems that were solved.

Another semester-long issue was that of article data consistency. The translation team was in charge of uploading data in the correct format. The team used a Google Spreadsheet, and often times, cells were left blank, not spell-checked, not format checked, etc. Given we indexed 200+ articles at a time, it was often impossible to locate the errors in the indexable file. Solr threw errors regarding incorrect formatting, but they were difficult to decipher. These persistent errors lead to us removing a few articles from the upload altogether. Discussions regarding formatting were held at weekly meetings, but the importance of formatting was difficult to project. A similar and frequent issue occurred with Solr's formatting of dates. The expected format was YYYY-MM-DDTHH:MM:SSZ, which was contrary to what was currently stored. We informed the project team to change the article dates to the correct format, but it took longer than expected to accomplish this task, and, in the meantime, we used demo data.

Once we indexed article data, we encountered search configuration issues. Full-text searches only returned case-sensitive exact matches. For example, when searching for articles written in Spanish, "spanish" did not return any results, but "Spanish" returned all articles with Spanish as their original text. We decided to add Solr Factories and Tokenizers to the schema.xml field types to combat this issue. The lowerCaseFilterFactory parsed all data as lowercase letters, which allowed case-insensitive results.

The final problem encountered involved a broken link from the search results page to a single document view. This problem has not yet been solved, but there is hope to have a fix by the end of the semester. Additional to meeting with a Ruby on Rails developer, we have posted a Blacklight user forum addressing the linking issue.

Our group has gained much knowledge this semester, the most obvious through working with novel applications. Another lesson includes starting assignments earlier even if all resources are not available. We also should have reached out to other groups working with similar applications to troubleshoot issues and ask questions. Another important lesson learned was that of communication. The ongoing formatting issues were hard to communicate to project members that did not have CS background knowledge. This information gap proved to inhibit our accomplishments, in that the formatting requests took more time than necessary to fix. Using examples and ensuring group members had a full understanding of the desired formatting changes would have helped expedite the process of indexing article data.

Acknowledgements

We would like to express our appreciation to Tom Ewing for all of his helpful input and feedback during the implementation of the web database created for his Russian Flu project. We would like to also thank him for providing all the information and content needed to make the website a more finished product.

We wish to thank various people for the contribution to this project. The translation team for being part of the Russian Flu project, and their hard work in providing us with the translations needed to fill the translations database. Without their hard work we would not be able to display all the useful information for the researchers that will be using the site we have created. Beth Lancaster for setting up the server for us, and helping us with its use.

Finally, a special thanks to our Professor Dr. Fox, for providing us a path to go with in our project, his helpful suggestions in resources we were not familiar with, and connecting us with the right people to get our server in place.

References

"Apache Lucene." *Apache Lucene*. The Apache Software Foundation, 2012. Web. March 2014. <http://lucene.apache.org/solr/tutorial.html>.

Beer, Chris. "Projectblacklight/blacklight." *GitHub*. Project Blacklight, Apr. 2014. Web. Mar. 2014. <https://github.com/projectblacklight/blacklight/wiki/Quickstart>.

"CSS Tutorial." *CSS Tutorial*. W3Schools, n.d. Web. 18 Apr. 2014. <http://www.w3schools.com/css/default.asp>.

"Fast Maps for the Web and Mobile." *MapTiler*. Klokan Technologies GmbH, n.d. Web. Mar. 2014. <http://www.maptiler.com/>.

Fox, E.A., Multimedia, Hypertext, and Information Access Professor, personal communication, May. 2014.

Lancaster, B., Sr. UNIX Systems/Database Administrator, personal communication, Apr. 2014.

"StoryMapJS." *StoryMap JS*. Northwestern University Knight Lab, n.d. Web. May. 2014. <http://storymap.knightlab.com/>.

"TimelineJS." *Timeline JS*. Northwestern University Knight Lab, n.d. Web. Apr. 2014. <http://timeline.knightlab.com/>.